Augmented Lagrangian Constraint Handling for CMA-ES—Case of a Single Linear Constraint

Asma Atamna, Anne Auger, Nikolaus Hansen

Augmented Lagrangian Constraint Handling for CMA-ES—Case of a Single Linear Constraint

Asma Atamna, Anne Auger, Nikolaus Hansen

Inria**
LRI (UMR 8623), University of Paris-Saclay, France

Abstract. We consider the problem of minimizing a function f subject to a single inequality constraint $g(\mathbf{x}) \leq 0$, in a black-box scenario. We present a covariance matrix adaptation evolution strategy using an adaptive augmented Lagrangian method to handle the constraint. We show that our algorithm is an instance of a general framework that allows to build an adaptive constraint handling algorithm from a general randomized adaptive algorithm for unconstrained optimization. We assess the performance of our algorithm on a set of linearly constrained functions, including convex quadratic and ill-conditioned functions, and observe linear convergence to the optimum.

1 Introduction

Evolution strategies (ESs) are derivative-free continuous optimization algorithms that are now well-established to solve unconstrained optimization problems of the form $\min_{\mathbf{x}} f(\mathbf{x})$, $f : \mathbb{R}^n \to \mathbb{R}$, where n is the dimension of the search space. The state-of-the-art ES, the covariance matrix adaptation evolution strategy (CMA-ES) [7], is especially powerful at solving a wide range of problems and particularly ill-conditioned problems [8,5]. It typically exhibits linear convergence. The default CMA-ES algorithm implements comma selection where the best solution is not preserved from one iteration to the next one (contrary to plus selection). Comma selection is an important feature of CMA-ES that entails robustness of the algorithm to various types of ruggedness including noise.

Linear convergence being a central aspect of an ES in the unconstrained case, a $(1 + 1)$-ES using an adaptive augmented Lagrangian constraint handing—to deal with a single inequality constraint—has been introduced in [3] with the motivation to obtain a linearly converging algorithm. Empirical results show the linear convergence of the algorithm on the sphere and moderately ill-conditioned ellipsoid functions, subject to one linear constraint. In [4], the authors present a variant of the previous $(1 + 1)$-ES with augmented Lagrangian constraint handling and study theoretically its linear convergence using a Markov chain approach. In both mentioned works, the step-size is adapted using the $1/5$th success rule [10] while the covariance matrix is fixed to the identity. On ill-conditioned problems, however, adapting the covariance matrix is crucial. It is hence natural to wonder whether it is possible to design a CMA-ES variant with augmented Lagrangian constraint handling. The algorithms presented in [3,4],

** Research centre Saclay–Île-de-France, TAO team, lastname@lri.fr

however, use plus selection and *can thus a priori not be used* directly to design such a variant.

In this context, we consider the constrained problem of minimizing $f : \mathbb{R}^n \to \mathbb{R}$ subject to a single inequality constraint $g(\mathbf{x}) \leq 0$, $g : \mathbb{R}^n \to \mathbb{R}$. More formally, we write

$$\min_{\mathbf{x}} f(\mathbf{x}) \quad \text{subject to} \quad g(\mathbf{x}) \leq 0 \ . \tag{1}$$

We bring to light that the algorithms previously presented in [3,4] derive from a more general framework that seamlessly allows to build an adaptive constraint handling algorithm from a general adaptive stochastic search method. We then naturally apply this finding to build a $(\mu/\mu_{\mathrm{w}}, \lambda)$-CMA-ES variant with adaptive augmented Lagrangian constraint handling. We opted for using the median success rule step-size adaptation (MSR) [2] because it is an extension of the $1/5$th success rule algorithm used in [3,4]. We then test the resulting algorithm—the $(\mu/\mu_{\mathrm{w}}, \lambda)$-MSR-CMA-ES with augmented Lagrangian constraint handling—on a set of functions, including convex quadratic as well as ill-conditioned functions, subject to one linear inequality constraint.

The rest of this paper is organized as follows: we introduce some basics about augmented Lagrangian in Section 2. Then, we define the general framework and apply it to the $(\mu/\mu_{\mathrm{w}}, \lambda)$-MSR-CMA-ES in Section 3. We present our empirical results in Section 4 and conclude with a discussion in Section 5.

Notations We introduce here the notations that are not explicitly defined in the rest of the paper. We denote $\mathbb{R}^+$ the set of positive real numbers and $\mathbb{R}^+_>$ the set of strictly positive real numbers. $\mathbb{N}_>$ is the set of natural numbers without 0. $\mathbf{x} \in \mathbb{R}^n$ is a column vector, $\mathbf{x}^\intercal$ is its transpose, and $\mathbf{0} \in \mathbb{R}^n$ is the zero vector. $\|\mathbf{x}\|$ denotes the Euclidean norm of $\mathbf{x}$ and $\sim$ equality in distribution. $(\mu/\mu_{\mathrm{w}}, \lambda)$ denotes comma selection with weighted recombination and $(1 + 1)$ denotes plus selection with one parent and one offspring. $\mathbf{I}_{n\times n} \in R^{n\times n}$ is the identity matrix. $\mathbf{x}_i$ is the ith component of vector $\mathbf{x}$. The derivative with respect to $\mathbf{x}$ is denoted $\nabla_{\mathbf{x}}$. Finally, $\mathbf{1}_{\{A\}}$ returns 1 if A is true and 0 otherwise.

2 Augmented Lagrangian Methods

Augmented Lagrangian methods are constraint handling approaches that transform the constrained optimization problem into an unconstrained one where an augmented Lagrangian is optimized [9,12].

The augmented Lagrangian consists of a Lagrangian $\mathcal{L}$ and a penalty function, with $\mathcal{L} : \mathbb{R}^{n+1} \to \mathbb{R}$ defined as

$$\mathcal{L}(\mathbf{x}, \gamma) = f(\mathbf{x}) + \gamma g(\mathbf{x}) \tag{2}$$

for the objective function f subject to one constraint $g(\mathbf{x}) \leq 0$, where $\gamma \in \mathbb{R}$ is the Lagrange factor. The Lagrangian encodes the KKT stationarity condition which states that, given some regularity conditions are satisfied (constraint qualifications), if $\mathbf{x}^* \in \mathbb{R}^n$ is a local minimum of the constrained problem, then there exists a constant $\gamma^* \in \mathbb{R}^+$, called the Lagrange multiplier, such that

$$\underbrace{\nabla_{\mathbf{x}} f(\mathbf{x}^*) + \gamma^* \nabla_{\mathbf{x}} g(\mathbf{x}^*)}_{\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}^*, \gamma^*)} = \mathbf{0} \ ,$$

where we assume here that f and g are differentiable at $\mathbf{x}^*$.

A penalty function is combined with the Lagrangian $\mathcal{L}$ to create the augmented Lagrangian h. There exist different ways to construct the augmented Lagrangian and we refer to [11] for a deeper discussion about this topic. In this work, we use the following augmented Lagrangian

$$h(\mathbf{x}, \gamma, \omega) = f(\mathbf{x}) + \begin{cases} \gamma g(\mathbf{x}) + \frac{\omega}{2} g^2(\mathbf{x}) & \text{if } \gamma + \omega g(\mathbf{x}) \geq 0 \\ -\frac{\gamma^2}{2\omega} & \text{otherwise} \end{cases} , \tag{3}$$

where $\omega > 0$ is a penalty factor. The same augmented Lagrangian was used for the first time within an ES in [3]. The function h is minimized successively with respect to $\mathbf{x}$, and γ and ω are updated so that γ approaches the Lagrange multiplier γ^* and ω favors feasible solutions. By adapting γ, the penalty factor ω does not have to grow to infinity to achieve convergence, unlike with quadratic penalty function methods [11].

Let $\mathbf{x}_{\text{opt}}$ be the optimum of the constrained problem in (1) and let γ_{opt} be the corresponding Lagrange multiplier. If f and g are differentiable at $\mathbf{x}_{\text{opt}}$, then for all $\omega > 0$,

$$\nabla_{\mathbf{x}} h(\mathbf{x}_{\text{opt}}, \gamma_{\text{opt}}, \omega) = \nabla_{\mathbf{x}} f(\mathbf{x}_{\text{opt}}) + \max(0, \gamma_{\text{opt}} + \omega g(\mathbf{x}_{\text{opt}})) \nabla_{\mathbf{x}} g(\mathbf{x}_{\text{opt}}) = \mathbf{0} \ .$$

3 A General Framework for Adaptive Augmented Lagrangian Constraint Handling

In [3] and [4], the authors present two $(1 + 1)$-ESs with an augmented Lagrangian constraint handling approach for the optimization problem in (1). The algorithms derive from a general framework for building a constraint handling adaptive algorithm. This framework starts with a randomized adaptive algorithm for minimizing an unconstrained function $f : \mathbb{R}^n \to \mathbb{R}$: the randomized adaptive algorithm can be identified by the sequence of its states $\mathbf{s}_t$ at iteration t that are iteratively computed from an update function $\mathcal{F}$ such that

$$\mathbf{s}_{t+1} = \mathcal{F}^f(\mathbf{s}_t, \mathbf{U}_{t+1}) \ , \tag{4}$$

where the superscript indicates the function being minimized and where $(\mathbf{U}_t)_{t \in \mathbb{N}_>}$ is a sequence of independent identically distributed (i.i.d.) random vectors. For instance, in the case of a $(1 + 1)$-ES in [3,4], the state is a vector of the search space (current estimate of the optimum) and a step-size.

We assume that the state $\mathbf{s}_t$ of the algorithm includes a vector $\mathbf{X}_t \in \mathbb{R}^n$ which typically encodes the current estimate of the optimum at iteration t. Note that the transition function $\mathcal{F}$ above includes a step where candidate solutions are sampled from the current state $\mathbf{s}_t$ and the random vector $\mathbf{U}_{t+1}$, and evaluated on the objective function f.

From the adaptive algorithm above, we construct an algorithm with adaptive constraint handling to take into account a single constraint in the following way: we add to the state of the algorithm two scalars γ_t and ω_t that correspond respectively to the Lagrange factor and the penalty factor of the augmented Lagrangian h at iteration t. Therefore, the state at iteration t is $\mathbf{s}_t{}' = [\mathbf{s}_t, \gamma_t, \omega_t]$. The objective function used at each iteration to evaluate a candidate solution $\mathbf{X}_{t+1}^i$ is now

$$h_{(\gamma_t, \omega_t)}(\mathbf{X}_{t+1}^i) := h(\mathbf{X}_{t+1}^i, \gamma_t, \omega_t) \ , \tag{5}$$

where h is the augmented Lagrangian defined in (3). Finally, the update of the state $\mathbf{s}_t{}'$ of the adaptive algorithm with augmented Lagrangian constraint handling takes place in two steps: first, $\mathbf{s}_t$ is updated via

$$\mathbf{s}_{t+1} = \mathcal{F}^{h_{(\gamma_t,\omega_t)}}(\mathbf{s}_t, \mathbf{U}_{t+1}) \ , \tag{6}$$

where candidate solutions are now evaluated on $h_{(\gamma_t,\omega_t)}$ instead of f. Then, the parameters γ_t and ω_t of h are updated. In [3], γ_t is updated according to

$$\gamma_{t+1} = \max(0, \gamma_t + \omega_t g(\mathbf{X}_{t+1})) \ , \tag{7}$$

while in [4], the authors use the following update

$$\gamma_{t+1} = \gamma_t + \omega_t g(\mathbf{X}_{t+1}) \ . \tag{8}$$

For ω_t, the following update is used in both [3] and [4]

$$\omega_{t+1} = \begin{cases} \omega_t \chi^{1/4} & \text{if } \omega_t g^2(\mathbf{X}_{t+1}) < k_1 \frac{|h(\mathbf{X}_{t+1},\gamma_t,\omega_t) - h(\mathbf{X}_t,\gamma_t,\omega_t)|}{n} \\ & \text{or } k_2 |g(\mathbf{X}_{t+1}) - g(\mathbf{X}_t)| < |g(\mathbf{X}_t)| \\ \omega_t \chi^{-1} & \text{otherwise} \end{cases} \ , \tag{9}$$

for some constants $\chi > 1$, $k_1, k_2 \in \mathbb{R}^+$.

Based on these examples, we introduce some general update functions $\mathcal{G}_\gamma$ and $\mathcal{G}_\omega$ for the updates of γ_t and ω_t defined implicitly via

$$\gamma_{t+1} = \mathcal{G}_\gamma^g((\gamma_t, \omega_t), \mathbf{X}_{t+1}) \tag{10}$$

$$\omega_{t+1} = \mathcal{G}_\omega^{(f,g)}((\mathbf{X}_t, \gamma_t, \omega_t), \mathbf{X}_{t+1}) \ . \tag{11}$$

The superscript in $\mathcal{G}_\gamma$ and $\mathcal{G}_\omega$ indicates that the function value is used in the update.

3.1 The $(\mu/\mu_\mathbf{w}, \lambda)$-MSR-CMA-ES with Adaptive Augmented Lagrangian

We now apply the general framework sketched above to the covariance matrix adaptation evolution strategy (CMA-ES) with median success rule step-size adaptation (MSR). We start by presenting the algorithm for the unconstrained case then we give the updates of the augmented Lagrangian parameters γ_t and ω_t.

The (unconstrained) CMA-ES with MSR The original CMA-ES with MSR is given in Algorithm 1, without the highlighted parts. The algorithm proceeds iteratively: at each iteration t, λ candidate solutions (offspring) $\mathbf{X}_{t+1}^i$, $i = 1, \cdots, \lambda$, are sampled according to Line 5, where $\mathbf{X}_t \in \mathbb{R}^n$ is the current estimate of the optimum (mean vector), $\sigma_t \in \mathbb{R}^+$ is the step-size, and $\mathbf{U}_{t+1}^i \in \mathbb{R}^n$, $i = 1, \cdots, \lambda$, are i.i.d. random vectors sampled from the normal distribution $\mathcal{N}(\mathbf{0}, \mathbf{C}_t)$, with mean $\mathbf{0} \in \mathbb{R}^n$ and covariance matrix $\mathbf{C}_t \in \mathbb{R}^{n \times n}$. The offspring are ordered according to their fitness (f-value in the unconstrained case) in Line 6, where $i : \lambda$ is the index of the ith best offspring. The μ best offspring (parents) are then recombined (Line 7) to create the new mean vector $\mathbf{X}_{t+1}$, where the weights $w_i > 0$, $i = 1, \cdots, \mu$, satisfy $w_1 > \cdots > w_\mu$ and $\sum_{i=1}^\mu w_i = 1$.

The step-sized σ_t is adapted in Lines 8 to 11 using the MSR step-size adaptation [2]. MSR is a success-based step-size adaptation method which extends the well-known $1/5$th success rule step-size adaptation [10], used with plus selection, to comma selection. The step-size is adapted depending on "success", where the success is defined as the median offspring $\mathbf{X}_{t+1}^{\boldsymbol{m}(\lambda)}$ (fitness-wise) of the current population being better than the jth best offspring $\mathbf{X}_t^{j:\lambda}$ of the previous population. In practice, we choose j to be the 30th percentile–the value for which the median success probability is roughly $1/2$ on the sphere function with optimal step-size [2]. The number K_{succ} of offspring better than $\mathbf{X}_t^{j:\lambda}$ is computed in Line 8. Note that $K_{\text{succ}} \geq \lambda/2$ is equivalent to $h(\mathbf{X}_{t+1}^{\boldsymbol{m}(\lambda)}, \gamma_t, \omega_t) \leq h(\mathbf{X}_t^{j:\lambda}, \gamma_t, \omega_t)$. Therefore, we define the success measure z_t in Line 9 such that $z_t \geq 0$ if and only if $\mathbf{X}_{t+1}^{\boldsymbol{m}(\lambda)}$ is successful. z_t is cumulated in q_{t+1} (Line 10) and, finally, σ_t is updated in Line 11: it increases in the presence of success ($q_{t+1} > 0$) and decreases otherwise in order to increase the probability of success.

The covariance matrix $\mathbf{C}_t$ is adapted with CMA [7] in Lines 12 and 13. The update is a combination of the so-called rank-one-update and rank-μ-update. A detailed discussion on CMA can be found in [6].

Finally, the jth best offspring is updated in Line 17. Therefore, the state of the algorithm in the unconstrained case is

$$\mathbf{s}_t = (\mathbf{X}_t, \sigma_t, q_t, p_t, \mathbf{C}_t, \mathbf{X}_t^{j:\lambda}) \ .$$

The constrained $(\mu/\mu_{\text{w}}, \lambda)$-MSR-CMA-ES with adaptive augmented Lagrangian As explained in the general framework, the fitness f is replaced with the augmented Lagrangian h in the constrained case. The parameters γ_t and ω_t are adapted in Lines 15 and 16 in Algorithm 1, where changes in comparison to the unconstrained case are highlighted in gray.

The Lagrange factor γ_t is adapted in Line 15. It is increased when the new solution $\mathbf{X}_{t+1}$ is unfeasible and decreased otherwise, unless it is zero. The derivation of this update is discussed in details in [11].

For the penalty parameter ω_t, we use the original update proposed in [3] for the $(1+1)$-ES with augmented Lagrangian. The update rule is given in Line 16. ω_t is increased either when (i) the augmented Lagrangian h does not change "enough" after γ_t and ω_t are updated to avoid stagnation. This is translated by the first inequality where

$$\omega_t g^2(\mathbf{X}_{t+1}) \approx |h(\mathbf{X}_{t+1}, \gamma_t + \Delta\gamma_t, \omega_t + \Delta\omega_t) - h(\mathbf{X}_{t+1}, \gamma_t, \omega_t)|$$

is compared to the change in h due to the change in $\mathbf{X}_t$, $|h(\mathbf{X}_{t+1}, \gamma_t, \omega_t) - h(\mathbf{X}_t, \gamma_t, \omega_t)|$. ω_t is also increased when (ii) the change in the value of the constraint function is not large enough (second inequality in Line 16). To prevent an unnecessary ill-conditioning of the problem, ω_t is decreased whenever conditions (i) and (ii) are not satisfied.

4 Empirical Results

We evaluate Algorithm 1 on the sphere function (f_{sphere}), two ellipsoid functions (f_{elli}) with condition numbers $\alpha = 10^2, 10^6$, f_{sphere}^2, $f_{\text{sphere}}^{0.5}$, the different powers function

Algorithm 1 $(\mu/\mu_{\mathrm{w}}, \lambda)$-MSR-CMA-ES with Augmented Lagrangian Constraint Handling

0 **given** $n \in \mathbb{N}_>$, $\chi = 2^{1/n}$, $k_1 = 3$, $k_2 = 5$, $\mu, \lambda \in \mathbb{N}_>$, $j = 0.3\lambda$, $0 \leq w_i < 1$, $\sum_{i=1}^{\mu} w_i = 1$,

$$\mu_{\text{eff}} = 1/\sum_{i=1}^{\mu} w_i^2,\ c_\sigma = 0.3,\ d_\sigma = 2 - 2/n,\ c_c = \frac{4 + \mu_{\text{eff}}/n}{n + 4 + 2\mu_{\text{eff}}/n}$$

$$c_1 = \frac{2}{(n+1.3)^2 + \mu_{\text{eff}}},\ c_\mu = \min\left(1 - c_1, 2\frac{\mu_{\text{eff}} - 2 + 1/\mu_{\text{eff}}}{(n+2)^2 + \mu_{\text{eff}}}\right)$$

1 **initialize** $\mathbf{X}_0 \in \mathbb{R}^n$, $\sigma_0 \in \mathbb{R}^+_>$, $\mathbf{C}_0 = \mathbf{I}_{n\times n}$, $t = 0$, $q_0 = 0$, $p_0 = \mathbf{0}$,
constrained_problem // **true** if the problem is constrained, **false** otherwise

2 **if** constrained_problem

3 **initialize** $\gamma_0 \in \mathbb{R}$, $\omega_0 \in \mathbb{R}^+_>$

4 **while** stopping criteria not met

5 $\mathbf{X}^i_{t+1} = \mathbf{X}_t + \sigma_t \mathbf{U}^i_{t+1}$, $\mathbf{U}^i_{t+1} \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_t), i = 1, \cdots, \lambda$ // sample candidate solutions

6 Extract indices $\{1:\lambda, \cdots, \lambda:\lambda\}$ of ordered candidate solutions such that

$$\begin{cases} h(\mathbf{X}^{1:\lambda}_{t+1}, \gamma_t, \omega_t) \leq \ldots \leq h(\mathbf{X}^{\lambda:\lambda}_{t+1}, \gamma_t, \omega_t) & \text{if constrained\_problem} \\ f(\mathbf{X}^{1:\lambda}_{t+1}) \leq \ldots \leq f(\mathbf{X}^{\lambda:\lambda}_{t+1}) & \text{otherwise} \end{cases}$$

7 $\mathbf{X}_{t+1} = \sum_{i=1}^{\mu} w_i \mathbf{X}^{i:\lambda}_{t+1} = \mathbf{X}_t + \sigma_t \sum_{i=1}^{\mu} w_i \mathbf{U}^{i:\lambda}_{t+1}$ // recombine μ best candidate solutions

8 $K_{\text{succ}} = \begin{cases} \sum_{i=1}^{\lambda} \mathbf{1}_{\{h(\mathbf{X}^i_{t+1}, \gamma_t, \omega_t) \leq h(\mathbf{X}^{j:\lambda}_t, \gamma_t, \omega_t)\}} & \text{if constrained\_problem} \\ \sum_{i=1}^{\lambda} \mathbf{1}_{\{f(\mathbf{X}^i_{t+1}) \leq f(\mathbf{X}^{j:\lambda}_t)\}} & \text{otherwise} \end{cases}$

9 $z_t = \frac{2}{\lambda}\left(K_{\text{succ}} - \frac{\lambda}{2}\right)$ // compute success measure

10 $q_{t+1} = (1 - c_\sigma) q_t + c_\sigma z_t$

11 $\sigma_{t+1} = \sigma_t \exp\left(\frac{q_{t+1}}{d_\sigma}\right)$ // update step-size

12 $p_{t+1} = (1 - c_c) p_t + \sqrt{c_c(2 - c_c)\mu_{\text{eff}}}\left(\frac{\mathbf{X}_{t+1} - \mathbf{X}_t}{\sigma_t}\right)$ // cumulation path for CMA

13 $\mathbf{C}_{t+1} = (1 - c_1 - c_\mu)\mathbf{C}_t + c_1 p_{t+1} p_{t+1}^\intercal + c_\mu \sum_{i=1}^{\mu} w_i \left(\frac{\mathbf{X}^i_{t+1} - \mathbf{X}_t}{\sigma_t}\right)\left(\frac{\mathbf{X}^i_{t+1} - \mathbf{X}_t}{\sigma_t}\right)^\intercal$
// update covariance matrix

14 **if** constrained_problem

15 $\gamma_{t+1} = \max(0, \gamma_t + \omega_t g(\mathbf{X}_{t+1}))$ // update Lagrange factor

16 $\omega_{t+1} = \begin{cases} \omega_t \chi^{1/4} & \text{if } \omega_t g^2(\mathbf{X}_{t+1}) < k_1 \frac{|h(\mathbf{X}_{t+1}, \gamma_t, \omega_t) - h(\mathbf{X}_t, \gamma_t, \omega_t)|}{n} \\ & \text{or } k_2 |g(\mathbf{X}_{t+1}) - g(\mathbf{X}_t)| < |g(\mathbf{X}_t)| \\ \omega_t \chi^{-1} & \text{otherwise} \end{cases}$ // update penalty factor

17 $\mathbf{X}^{j:\lambda}_{t+1} = \mathbf{X}_t + \sigma_t \mathbf{U}^{j:\lambda}_{t+1}$ // update jth best solution

18 $t = t + 1$

($f_{\text{diff\_pow}}$), and the Rosenbrock function (f_{rosen}), with one linear inequality constraint. The functions are defined in Table 1. We consider the case where the constraint is active at the optimum $\mathbf{x}_{\text{opt}}$, i.e. $g(\mathbf{x}_{\text{opt}}) = 0$. We choose the optimum to be at $\mathbf{x}_{\text{opt}} = (10, \cdots, 10)^\intercal$ and construct the constraint function, $g(\mathbf{x}) = \mathrm{b}^\intercal \mathbf{x} + c$, so that the KKT stationarity condition is satisfied at $\mathbf{x}_{\text{opt}}$ with $\gamma_{\text{opt}} = 1$. Therefore,

$$\mathrm{b} = -\nabla_{\mathbf{x}} f_{\cdot}(\mathbf{x}_{\text{opt}})^\intercal \quad \text{and} \quad c = \nabla_{\mathbf{x}} f_{\cdot}(\mathbf{x}_{\text{opt}}) \mathbf{x}_{\text{opt}} \ ,$$

for each function. Note that all considered functions are differentiable at $\mathbf{x}_{\text{opt}} = (10, \cdots, 10)^\intercal$.

Name	Definition	Name	Definition
$f^{\alpha}_{\text{sphere}}(\mathbf{x})$	$\left(\frac{1}{2}\sum_{i=1}^{n} \mathbf{x}_i^2\right)^{\alpha}$	$f_{\text{diff\_pow}}(\mathbf{x})$	$\sqrt{\sum_{i=1}^{n} \lvert\mathbf{x}_i\rvert^{2+4\frac{i-1}{n-1}}}$
$f_{\text{elli}}(\mathbf{x})$	$\frac{1}{2}\sum_{i=1}^{n} \alpha^{\frac{i-1}{n-1}} \mathbf{x}_i^2$	$f_{\text{rosen}}(\mathbf{x})$	$\sum_{i=1}^{n-1} \left(10^2(\mathbf{x}_i^2 - \mathbf{x}_{i+1})^2 + (\mathbf{x}_i - 1)^2\right)$

Table 1: Definitions of the tested functions, where $f_{\text{sphere}} := f^{1}_{\text{sphere}}$.

For the step-size and the covariance matrix adaptation, we use the Python implementation of CMA-ES whose source code can be found at [1], with the default parameter setting detailed in [6]. We run the algorithm 11 times in $n = 10$, with $\mathbf{X}_0$ sampled uniformly in $[-5, 5]^n$, $\sigma_0 = 1$, $\gamma_0 = 5$, and $\omega_0 = 1$. The results are presented for one run in Figures 1 (f_{sphere}, f^{2}_{sphere}, and $f^{0.5}_{\text{sphere}}$) and 2 (f_{elli} with $\alpha = 10^2, 10^6$, $f_{\text{diff\_pow}}$, and f_{rosen}). On the left column of each figure are graphs of the evolution of the distance to the optimum $\|\mathbf{X}_t - \mathbf{x}_{\text{opt}}\|$, the step-size σ_t, the distance to the Lagrange multiplier $|\gamma_t - \gamma_{\text{opt}}|$, and the penalty factor ω_t in log-scale. On the right column of the figures are graphs representing the evolution of the coordinates of the mean vector $\mathbf{X}_t$.

Graphs on the right column of Figures 1 and 2 show the overall convergence of the algorithm to $\mathbf{x}_{\text{opt}}$. We also observe linear convergence of $\mathbf{X}_t$ to $\mathbf{x}_{\text{opt}}$, as well as linear convergence of γ_t to γ_{opt} and σ_t to 0 (left column of Figures 1 and 2). Moreover, $\|\mathbf{X}_t - \mathbf{x}_{\text{opt}}\|$, $|\gamma_t - \gamma_{\text{opt}}|$, and σ_t decrease at the same rate. On the other hand, the penalty factor ω_t is observed to converge to a stationary value after a certain number of iterations. We sometimes observe a stagnation in graphs of $\|\mathbf{X}_t - \mathbf{x}_{\text{opt}}\|$ due to numerical precision.

The largest convergence rate (when excluding the initial adaptation phase) is observed on f_{sphere} and the smallest one on $f^{0.5}_{\text{sphere}}$, where there is a factor of approximately 1.5 between the two convergence rates. However, there is some variance in the empirical convergence rate. In particular, on 11 performed runs we observe the highest variance in the empirical convergence rate for f_{elli} with $\alpha = 10^6$, $f_{\text{diff\_pow}}$, and f_{rosen}.

On f_{elli} with $\alpha = 10^6$, $f_{\text{diff\_pow}}$, and f_{rosen}, we observe a stagnation of $\mathbf{X}_t$ in the early stages of the algorithm (left column in Figure 2). The reason is that the adaptation of the covariance matrix takes longer on ill-conditioned problems. This explains the slow convergence of some coordinates of $\mathbf{X}_t$ to 10 (right column in Figure 2). Once the covariance matrix is adapted, the convergence occurs.

When comparing 11 single runs of Algorithm 1 to the $(1+1)$-ESs with augmented Lagrangian in [3,4] (not shown for space reasons) on constrained f_{sphere}, f_{elli} (in $n =$

10), it appears that on f_{sphere}, Algorithm 1 needs approximately up to 1.5 times more function evaluations than algorithms in [3,4] to reach a distance to the optimum of 10^{-4}. On f_{elli} with $\alpha = 10^2$, however, Algorithm 1 is faster and needs approximately 1.3 times less function evaluations to reach the same distance, with $\alpha = 10^6$, Algorithm 1 is around 167 times faster to reach a target of 15 (this large difference is due to the adaptation of the covariance matrix).

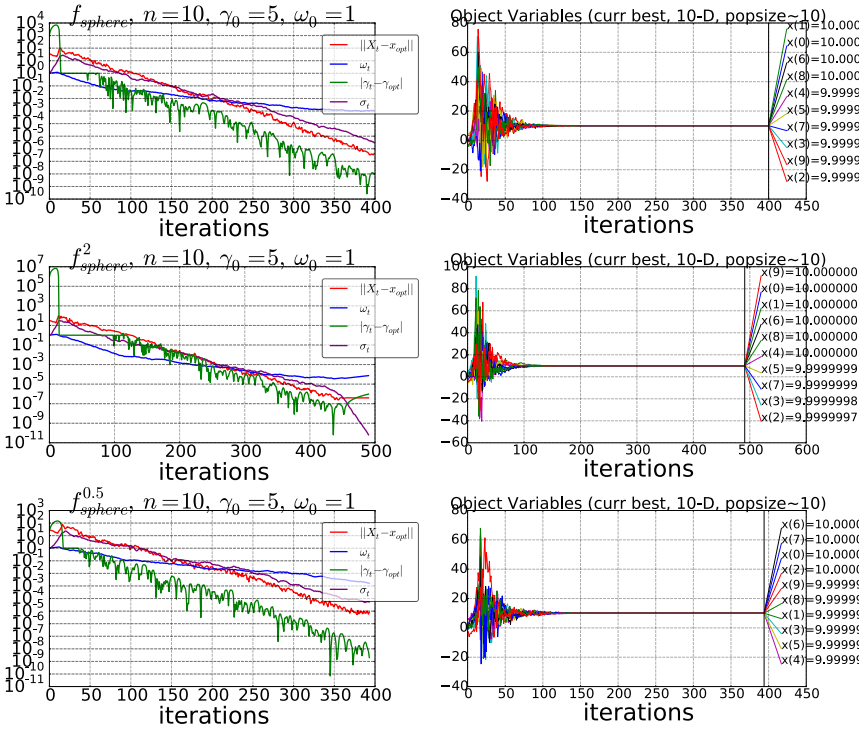

Fig. 1: Single runs of $(\mu/\mu_{\text{w}}, \lambda)$-MSR-CMA-ES with augmented Lagrangian on f_{sphere} (top row), f^2_{sphere} (middle row), and $f^{0.5}_{\text{sphere}}$ (bottom row) in $n = 10$. The optimum $\mathbf{x}_{\text{opt}} = (10, \cdots, 10)^\intercal$. Left: evolution of the distance to the optimum, the distance to the Lagrange multiplier, the penalty factor, and the step-size in log-scale. Right: evolution of the coordinates of $\mathbf{X}_t$.

5 Discussion

Linear convergence is a key aspect of ESs in both unconstrained and constrained optimization scenarios. As stated in [3], the minimum requirement for a constraint handling

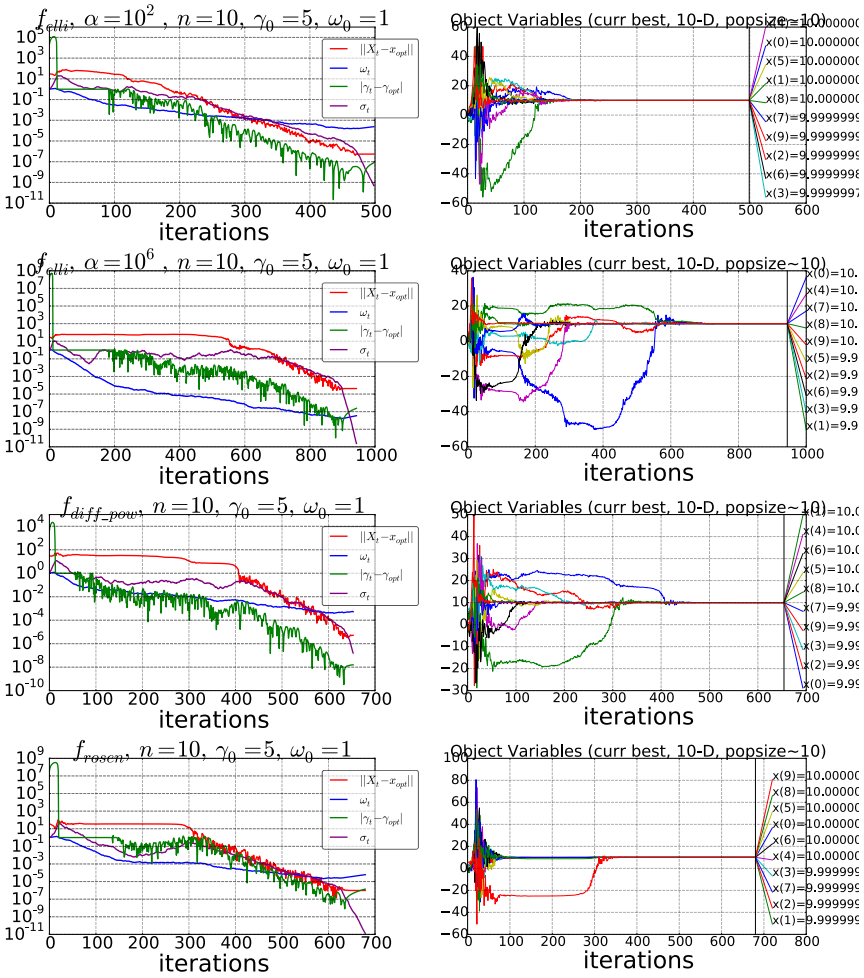

Fig. 2: Single runs of $(\mu/\mu_{\mathrm{w}}, \lambda)$-MSR-CMA-ES with augmented Lagrangian on f_{elli} with $\alpha = 10^2$ (first row), f_{elli} with $\alpha = 10^6$ (second row), $f_{\mathrm{diff\_pow}}$ (third row), and f_{rosen} (fourth row) in $n = 10$. The optimum $\mathbf{x}_{\mathrm{opt}} = (10, \cdots, 10)^{\mathsf{T}}$. Left: evolution of the distance to the optimum, the distance to the Lagrange multiplier, the penalty factor, and the step-size in log-scale. Right: evolution of the coordinates of $\mathbf{X}_t$.

ES is to converge linearly on convex quadratic functions with a single linear constraint. On the other hand, an algorithm for constrained optimization should be able to tackle ill-conditioned problems. Having that in mind, we proposed a $(\mu/\mu_{\mathrm{w}}, \lambda)$-CMA-ES with an augmented Lagrangian approach for handling one inequality constraint, where the choice of the augmented Lagrangian constraint handling was motivated by the promising results of its implementation for the $(1 + 1)$-ESs with $1/5$th success rule in [3,4].

Moreover, we showed that our algorithm–as well as $(1 + 1)$-ESs with augmented Lagrangian constraint handling in [3,4]–is an instance of a more general framework for building an adaptive constraint handling algorithm from a general adaptive algorithm for unconstrained optimization.

Experiments on linearly constrained convex quadratic functions, as well as ill-conditioned functions (including the ellipsoid and Rosenbrock functions), showed linear convergence of our algorithm to the unique optimum of the constrained problem.

Acknowledgments This work was supported by the grant ANR-2012-MONU-0009 (NumBBO) of the French National Research Agency.

References

1. https://pypi.python.org/pypi/cma. Python source code of CMA-ES.
2. O. Ait Elhara, A. Auger, and N. Hansen. A median success rule for non-elitist evolution strategies: Study of feasibility. In *Genetic and Evolutionary Computation Conference*, pages 415–422. ACM Press, 2013.
3. D. V. Arnold and J. Porter. Towards au Augmented Lagrangian Constraint Handling Approach for the $(1 + 1)$-ES. In *Genetic and Evolutionary Computation Conference*, pages 249–256. ACM Press, 2015.
4. A. Atamna, A. Auger, and N. Hansen. Analysis of Linear Convergence of a $(1 + 1)$-ES with Augmented Lagrangian Constraint Handling. To appear in the proceedings of the Genetic and Evolutionary Computation Conference, 2016.
5. A. Auger, N. Hansen, J. Perez Zerpa, R. Ros, and M. Schoenauer. Experimental Comparisons of Derivative Free Optimization Algorithms. In Jan Vahrenhold, editor, *8th International Symposium on Experimental Algorithms*, volume 5526, pages 3–15. Springer, 2009.
6. N. Hansen. The CMA Evolution Strategy: A Tutorial. http://arxiv.org/pdf/1604.00772v1.pdf, 2016.
7. N. Hansen and A. Ostermeier. Completely Derandomized Self-Adaptation in Evolution Strategies. *Evolutionary Computation*, 9(2):159–195, 2001.
8. N. Hansen, R. Ros, N. Mauny, M. Schoenauer, and A. Auger. Impacts of Invariance in Search: When CMA-ES and PSO Face Ill-Conditioned and Non-Separable Problems. *Applied Soft Computing*, 11:5755–5769, 2011.
9. M. R. Hestenes. Multiplier and Gradient Methods. *Journal of Optimization Theory and Applications*, 4(5):303–320, 1969.
10. S. Kern, S. D. Müller, N. Hansen, D. Büche, J. Ocenasek, and P. Koumoutsakos. Learning Probability Distributions in Continuous Evolutionary Algorithms - A Comparative Review. *Natural Computing*, 3(1):77–112, 2004.
11. J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, 2nd edition, 2006.
12. M. J. D. Powell. A Method for Nonlinear Constraints in Minimization Problems. In R. Fletcher, editor, *Optimization*, pages 283–298. Academic Press, 1969.